

RasPi

DESIGN
BUILD
CODE

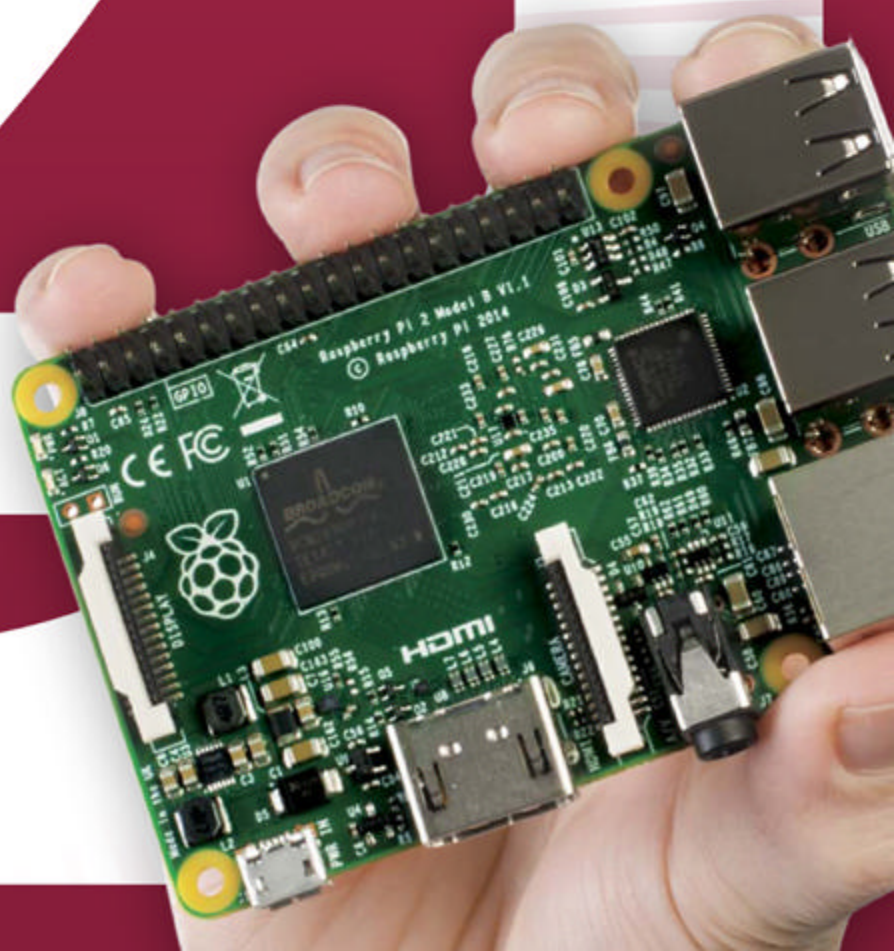
7

Get hands-on with your Raspberry Pi

Exclusive: Announcing the RASPBERRY PI

PLUS
SLO-MO
VIDEO

Everything
you need to know





Welcome



That's right – the Raspberry Pi 2 has arrived! We go behind the scenes at Pi Towers and speak to Pi 2 creators Eben Upton and

James Adams, who reveal everything you need to know about this powerful new board and give us the exclusive on some upcoming add-ons. And as usual, we've got plenty of projects for you to get stuck into as well – we're taking a good look at servers this time. From a simple file server designed to keep your private files out of third-party cloud silos like Dropbox or something more ambitious, like a VoIP server to enable a building full of people to phone each other over a local network, your Raspberry Pi is perfectly suited for the job. Have fun setting up!

Gavin Thomas

Deputy Editor

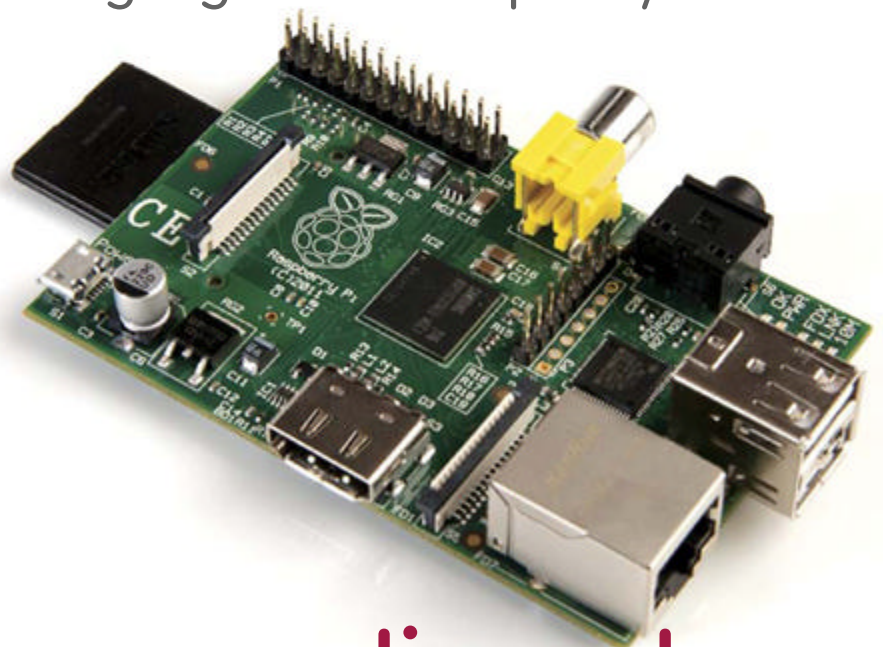
From the makers of
Linux User
& Developer

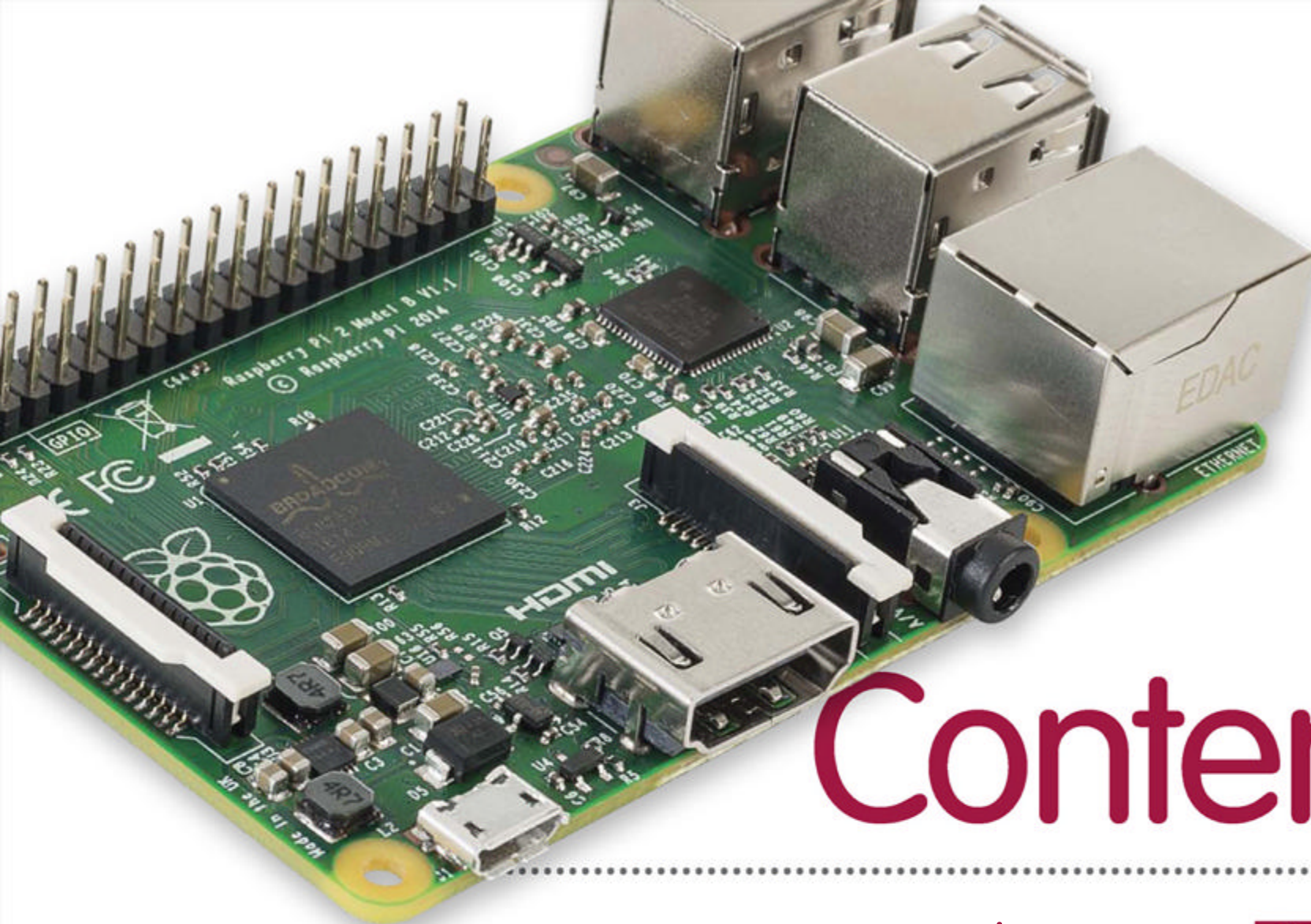
Join the conversation at...

@linuxusermag

Linux User & Developer

RasPi@imagine-publishing.co.uk





Contents

Raspberry Pi 2

It's here! Get the inside story on the new board



Browse privately with Onion Pi

Turn your Pi into a secure portable router



Pi2D2

Meet the hacked R2D2 robot with a Pi brain



Make a complete media centre

Set up your Pi HTPC and connect to it remotely



Build a VoIP server

Make phone calls over your network



Set up a file server

Make a personal cloud to safely share your content



Record slow-motion video

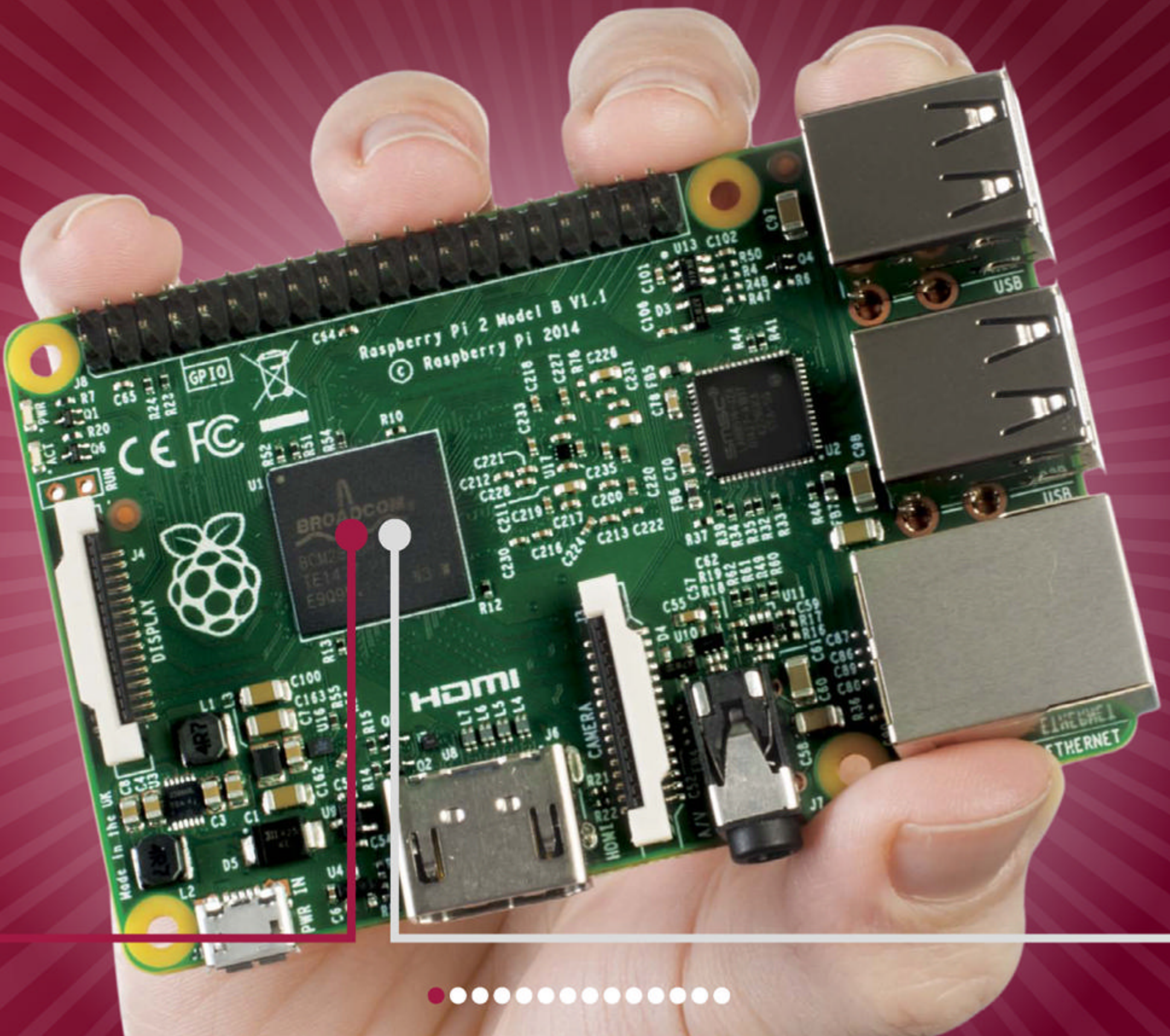
Slow down your shoots to 33% speed





Announcing the... Raspberry Pi 2

The Raspberry Pi 2 is here! We speak to Eben Upton and James Adams to get the inside story on the Pi 2 and the Raspberry Pi Foundation's 2015 plans



engineering and software teams who have developed the expanding family of Raspberry Pi products.

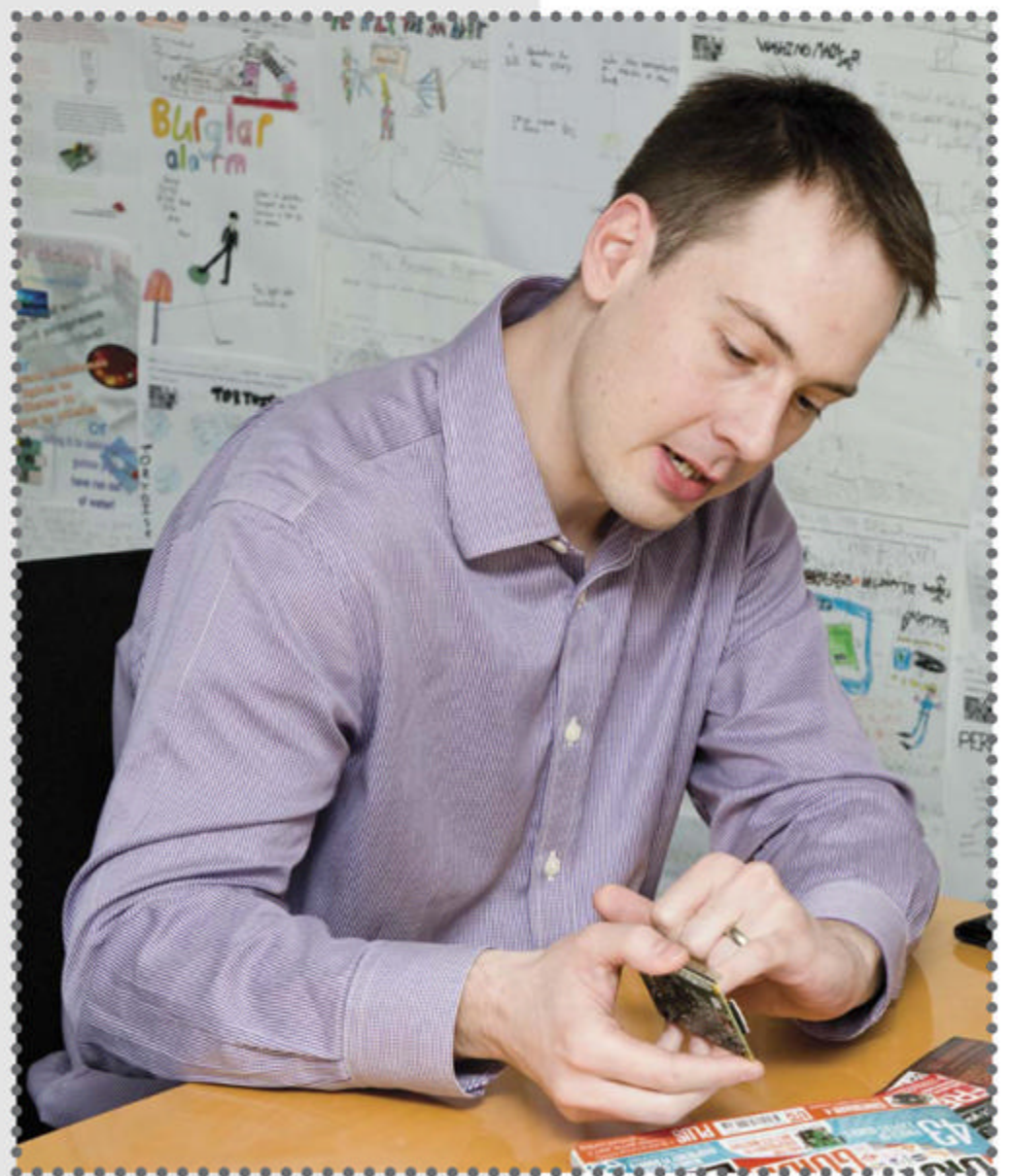
We spoke to James Adams, director of hardware at the Foundation, who explained how he first got involved in the Pi project: "I joined Raspberry Pi back in the beginning of 2013. So the original Raspberry Pi A and B weren't designed by me – that was Pete Lomax – and basically I came on board at the point where this group of guys were doing it all in their spare time. It sold well, there was some money, and it was obvious that they needed proper engineering resources, so we needed to hire some people, pay them salaries – set up shop properly. I was one of the two guys that they first hired – I knew Eben from way back – we used to work at Broadcom together and we've done lots of things from chip design to circuit board design, and in fact we worked together on the Video Core chip in the Raspberry Pi, the 3D graphics. Since I joined I've designed the B+, the A+, the Compute Module and now the Pi 2. The idea was to polish the thing and have a nice end-to-end, more coherent offering. And also try and fix some of the issues people had with the Pi."

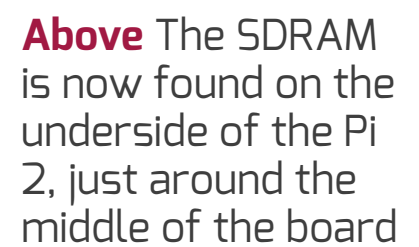
B+ built for Pi 2

The Foundation team is always on the raspberrypi.org forums and they get a lot of feedback, so they knew quite early on what people were happy with, what they weren't and also which

“Since I joined I’ve designed the B+, the A+, the Compute Module and now the Pi 2”

Below James took us through the new Pi 2 component by component

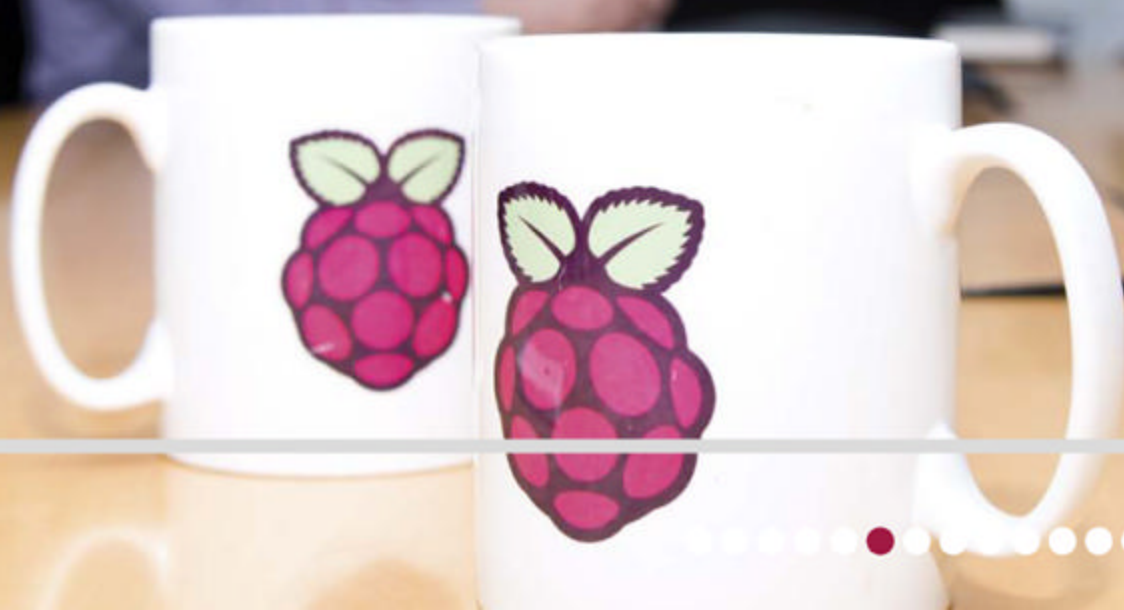


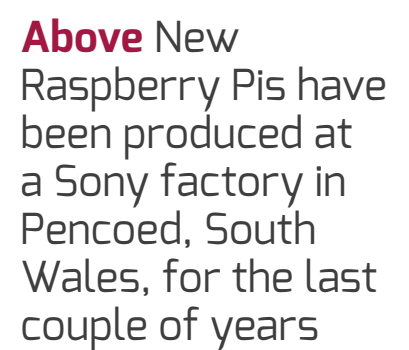


This was the decision to abandon the package-on-package design (PoP) used in the Raspberry Pi B+ and the original Models A and B. PoP is the process of stacking two or more packages on top of each other – in the Pi’s case, the processor and the SDRAM – with an interface to route signals between them. James says it was used partly because that’s what was sold to other customers for that chip in Broadcom, and partly because “it’s basically the best possible situation – you’re not quite sticking two chips together, but there’s very little in the way between the two that can go wrong. And there’s less routing on the board because you don’t have to route out this big SDRAM interface”. So why the change? “The way the BCM2836 was designed,” explains James, “we took the old 2835 chip, took the ARMv6 core out and we put in the quad-core ARMv7 with more cache just on the side. The rest of the chip, apart from that ARM complex, is basically identical. And the die grew, so it couldn’t physically fit in the PoP package. We wanted it to, but the original 2835 only just managed to fit in there and we’ve grown the chip size a little bit since then.”

“We took the old 2835 chip, took the ARMv6 core out and we put in the quad-core ARMv7”

Below Pi Towers boasts some wonderfully friendly mugs. And some nice caffeine vessels.





The new BCM2836 processor is, of course, the defining element of the Pi 2 – it's this chip that's boosting the Raspberry Pi's power by a factor of six. We asked James how much of its design was custom and how much was based on its predecessor: "It's very much based on the 2835," he replies. "So the story with the 2836 is that back when I joined, there was talk about what we'd do with the next Pi. Do we do anything with the next Pi? What is the Raspberry Pi? We decided that the Raspberry Pi is this \$35 form factor – we wanted a chip that would be better but that would also be the same, if that makes sense, and the way to do that is to custom-make one. Obviously Eben still works for Broadcom and the deal with Broadcom was that we'd put up some engineering resource, which was me, and there were a few other people in Broadcom working on this chip, so I went in for nine months for some contracting work. Basically we resurrected the design –

took the old chip, took the ARM core out, put the new ARM core in, tested it, simulated it, built it, and now we've got products based on it. You don't normally do chips this way, so it's a bit of an interesting one."

Upgrading to ARMv7

All the previous Raspberry Pi models have been based on the ARMv6 instruction set, so what was the motivation for the upgrade? James told us that “having the v7 instruction set is really important because a lot of people don’t really think about ARMv6 or compile for ARMv6. With Debian, for example, they don’t have native ARMv6 – they’re all focused on ARMv7 and now ARMv8. So even back when the Raspberry Pi came out, ARMv6 was a little bit of an issue because we didn’t just get an off-the-shelf distribution – we had to recompile everything for ARMv6. So that’s a great win – clock for clock, the core’s faster. We’ve upped the Megahertz, you’ve got four cores and it’s also got half a Meg of dedicated cache for the processor, so it actually makes a significant difference to speed. You have also got double RAM. So this is a very usable little desktop machine now.” Regarding the Megahertz, while the official documentation records a benchmark of 800MHz for the quad-core, in their own tests the Foundation’s engineering team actually got results that were approaching 900MHz.



Above Eben says the Pi 2 is now a useable desktop computer

“We’ve upped the Megahertz, you’ve got four cores and its got half a Meg of dedicated cache. You have also got double RAM”

With all the extra horsepower, you'd think that the power draw of the Raspberry Pi 2 would be significantly higher than with the Model B+. James reassured us, however, that this is not the case: "When you're idling and doing light tasks, the power consumption's about the same as the B+, but if you load the processor heavily (this is in extreme scenarios) then it can draw quite a bit more power, obviously, because you've got a bigger silicon area there consuming it. On the flip side, this is now non-PoP – there's more solder holding the physical chip down – so the heat dissipates better. We're not expecting anyone to see any real difference though. You might see some higher peak current draws but it's still within the envelope of the B+."

We also confirmed that the Raspberry Pi 2 is going to be fully backwards compatible and can be used inside your existing projects once you've got it set up: "You need a v7 kernel and that's it," explains Eben. "You don't even need new firmware because the firmware we've been shipping for the last three or four months already supports 2836." (We raised some eyebrows at this point.) "Once we had sample boards it was convenient to just put it in the firmware – it's just been lurking there. It's actually worth googling for BCM2836 – there's a pastebin with the kernel. It's possible we got rid of it but anonymous pastebins are searchable, and this thing had been sitting there for four or five months. It's got everything – it tells you what the cores are, how fast they are, and it's just been sitting there."

Rather than recompile the entirety of Raspbian to be ARMv7, the Foundation decided to ship the operating system with both the v6 and the v7 kernels, and on boot Raspbian will detect whether it's a Pi 2 or an older Pi

New Display module

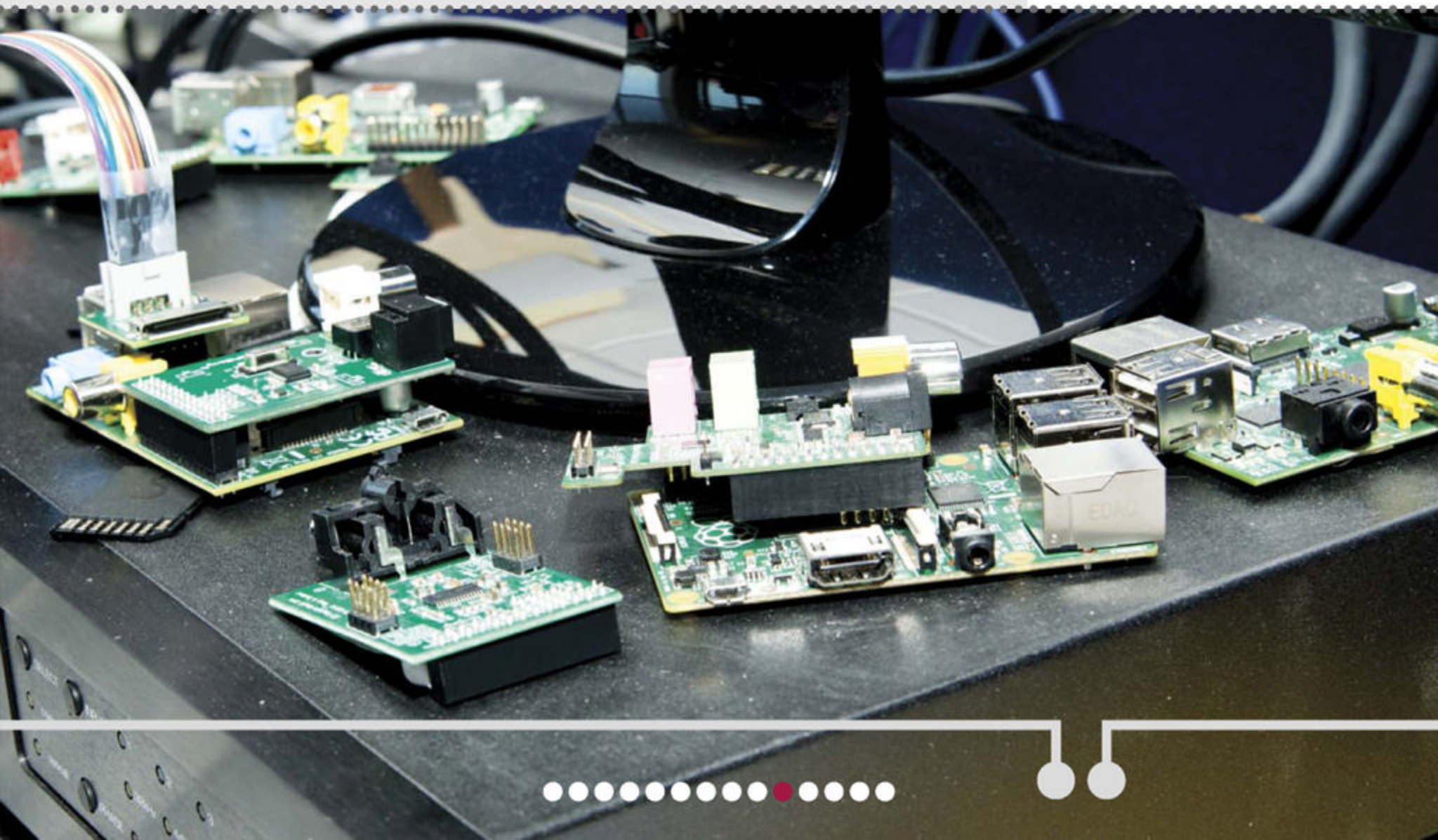
We caught wind of a few new products that are currently in development – among them the Raspberry Pi display module. A touchscreen designed for the Pi's MIPI DSI connector, the idea is that an adapter board will fit underneath the Pi and the screen over the top of it, so that the IO pins are accessible round the back and you can place the entire setup into an enclosure or other project.

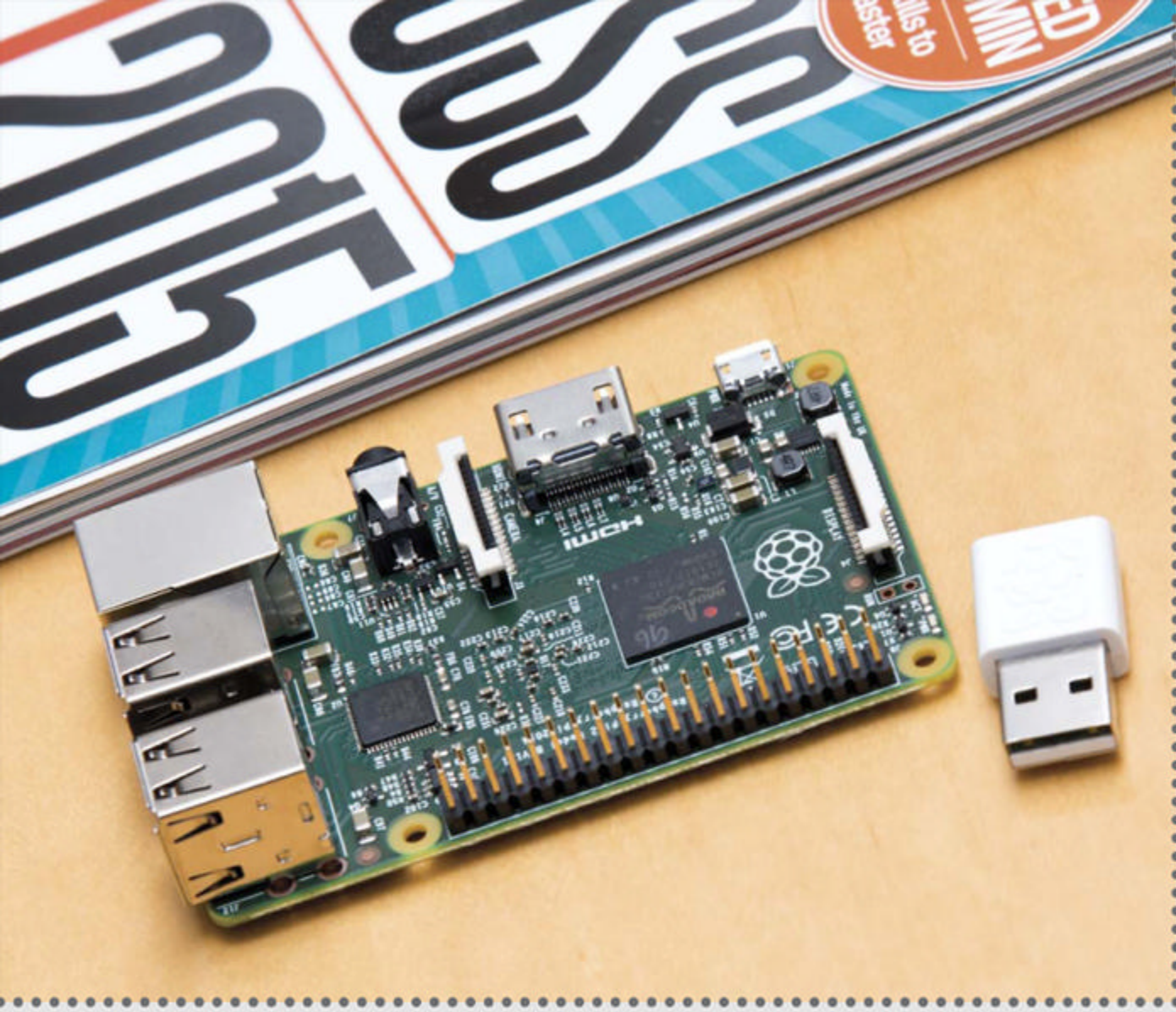
"It's a wide VGA, 800 x 480," says Eben, "but it's an industrial quality display so it has a very good viewing angle. This has got 10-point capacitive touch on the front as well."



that's being used and then run the required kernel – meaning that we won't need two different versions of Raspbian to account for the different requirements of the Pi 2 and its predecessors. As well as the twin kernels, Eben tells us that we should notice a real improvement to the design of Raspbian: "All open source software packages have very option-rich user experiences – they're designed for power users – and, leaving the configurability there in the configuration files, it's possible to remove a lot of the clutter away from the default configuration. So it's really been about taking stuff out – and also rationalising, making things work more consistently. So you've got a highlight colour on the window frame, for example – have that be the same highlight colour that's in the launch bar. Make all the fonts are the same. Use a nicer font. There's lots of that sort of attention-to-detail work that's been going in, and that's a thing we've been able to do as we've grown the number of engineers."

Below There were some wonderful jumbles of tech around Pi Towers – although a few goodies were quickly swept out of sight...





Model A and Compute Module

With Raspbian getting an exterior polish to match its engine upgrade and the Pi 2 set to smoothly replace the Model B+, we wondered what the next step was for the Foundation in terms of growing the Pi family. We asked Eben if a Raspberry Pi 2 Model A is on the cards: "So the issue is that I don't think I could do a Raspberry Pi 2 Model A at twenty bucks," he explained. "We're pretty attached to the \$20 price point – it's hard to do that with the improved SBC, not least because if you put a quad-core SBC down with a quarter of a Gig of RAM, it feels like a kind of imbalanced system. You probably want to put at least half a Gig of RAM on there and then you've got something where fitting that into \$20 is challenging. And most of the people using the A+ don't need the quad-core performance. We've gotten an awful lot of performance out of the ARM11 on the Raspberry Pi 1 and

New Wi-fi dongle

When we asked if onboard Wi-Fi was considered during the design process of the Raspberry Pi 2, Eben Upton quickly dashed out to fetch another brand new product from the Foundation by way of reply – a 150-Megabit Raspberry Pi Wi-Fi dongle.

The problem is that onboard Wi-Fi is quite expensive and getting the quality right without pushing the Pi above \$35 is a real challenge. The Foundation's solution is to offer its own high quality, offboard Wi-Fi.

“We’re shooting for something ridiculously low like \$5 plus VAT,” says Eben. “So with VAT, something like £4.”

A man with a friendly expression, wearing a grey zip-up sweater over a white collared shirt, stands in front of a wall densely decorated with various posters and diagrams related to the Raspberry Pi. The posters include titles like 'Pushing mixer!', 'Raspberry Pi', 'The Self-Driving House', 'Burglar Alarm', and 'Pi'. One prominent poster features a Raspberry Pi logo and the text 'Pi'. The man is gesturing with his right hand, palm up, towards the posters. The entire image is framed with a decorative border of small raspberries.

Above There's a whole wall dedicated to creative projects sent in by Pi enthusiasts

“The Raspberry Pi is a device for children to own themselves and have at home”

software that you can run on school computers. The nice thing is that most of these pieces of educational software are open source – even the ones you’re using under Windows at school – and you can use them on the Pi.”

Mission accomplished?

With the sea-change that has come about in the UK education system as a result of the new IT curriculum, things are looking very good indeed for the future of computing. Towards the end of the day, we were chatting to Eben about how rewarding it will be a few years down the line, when all the kids who are coding Sonic Pi and Scratch in schools right now start to hit college and university, and start pushing those computer science graduate numbers back up. Apparently, we don’t even need to wait that long: “The canary in the coal mine for us was the application numbers at Cambridge,” says Eben. “They had this massive decline from the Dotcom boom, like five or six hundred people applying for eighty places, down to 250 people in 2008. Interesting thing is, it’ll take a while to feed through kids who are playing with Raspberry Pis but it takes no time at all for it to feed through the message that computer science is important, valid. One thing we’ve made is this educational tool, but the other thing is a lot of noise. Actually, the noise has already taken effect – the application numbers now are above where they were in the Dotcom boom.

“At 2008 it was like a dead cat bounce,” elaborates Eben, “then it went up to seven or eight hundred. The nice thing is it’s actually loads of organisations who woke up going ‘We’ve got a problem!’ It catalysed itself – it was this wave of enthusiasm. A number of organisations materialised at exactly the same time around exactly the same thing, not competing with each other but just doing different aspects of the same thing. So in my

Astro Pi ready to launch

Later this year, two Raspberry Pis are going to be blasted off into low Earth orbit on the International Space Station, along with a brand new Raspberry Pi HAT (hardware attached on top) board called the Astro Pi.

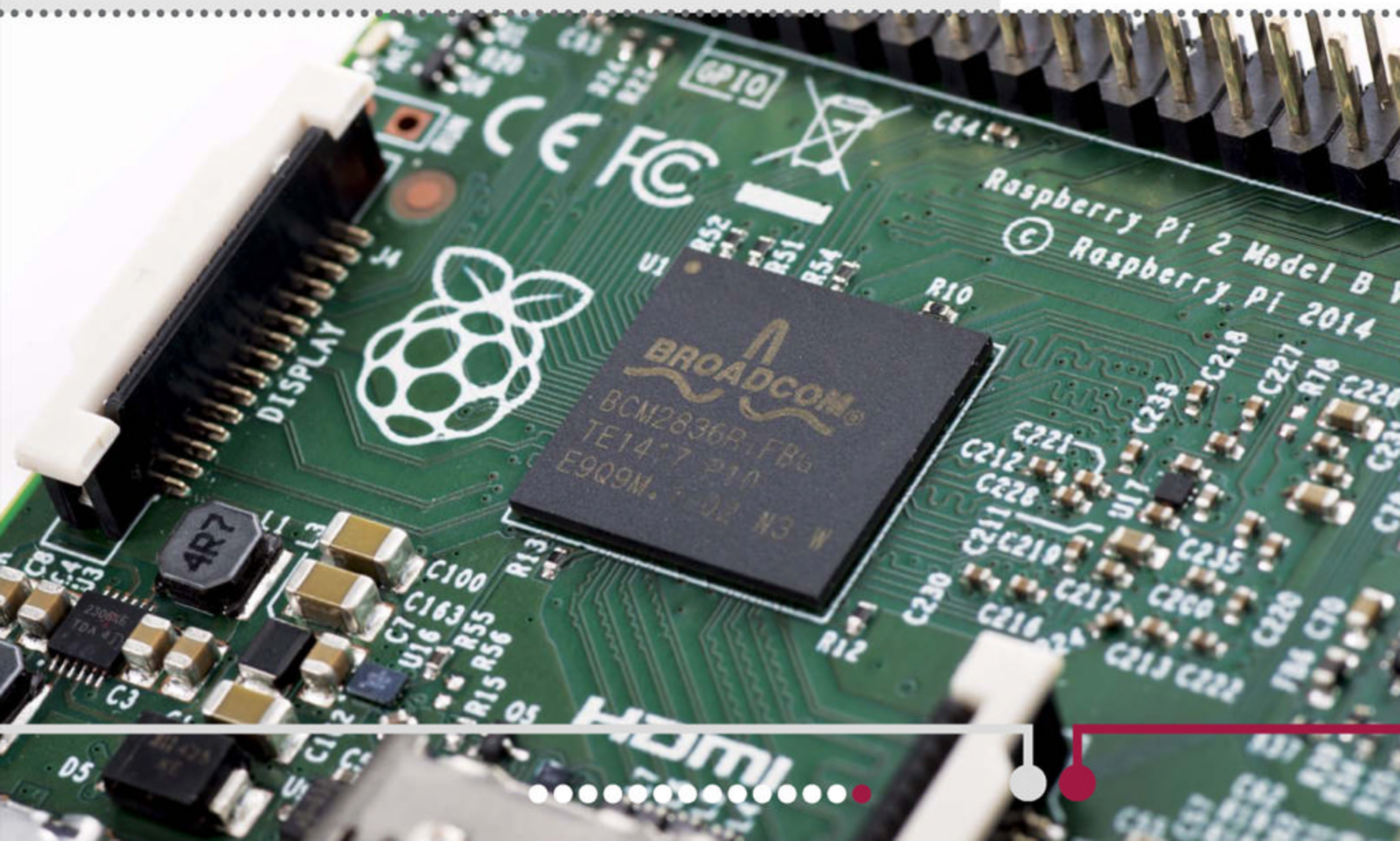
“I’ve just been putting the finishing touches on the board that’s going into Astro Pi,” says James. “It has this LED matrix you can use like a framebuffer, a little joystick, and an accelerometer, gyroscope, magnetometer, humidity, pressure and temperature sensors. We’ll try and make it available as cheaply as we can so people can buy it for fun or education.”



college interviews, when I was director of studies and we were interviewing people, and we had six credible applicants for three places, it was just horrific. My college interviewed 42 applicants this year – same college, just ten years later – again for three places. With 42 applicants you get a few good ones, and then you go and beg to the administration and say, ‘Look, just let me pick five or six.’ But yeah, I was talking to my friend who was interviewing and he said, ‘We just interviewed for days.’ It was only an afternoon when I was doing it!”

It’s an amazing response and the Raspberry Pi Foundation is well on its way to completely transforming computing education. On a final note, Eben reassured us that his ambitions don’t stop there: “We still want the desktop. Linus wants the desktop and we want to help him get it. There’s no better way to get him the desktop than to make it very, very cheap to get the hardware.” Reader, you’ll be the first to know once we’ve replaced each of our ageing rigs with a powerful little Pi. We can’t wait!

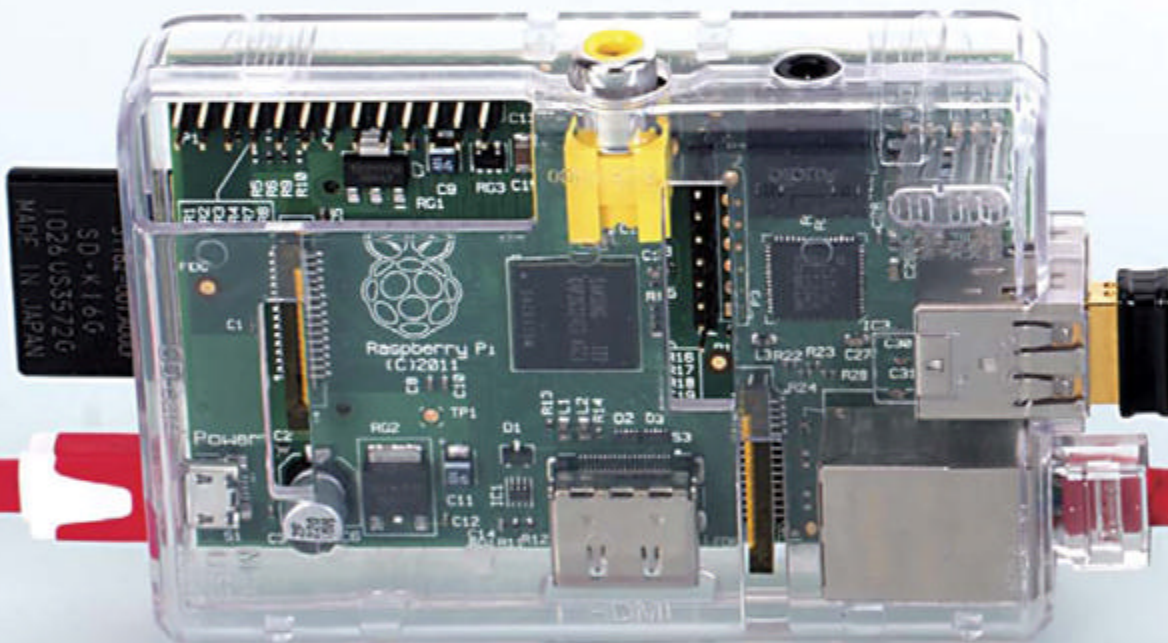
Below Raspberry Pi 2 and the new Raspbian are available now – see raspberrypi.org



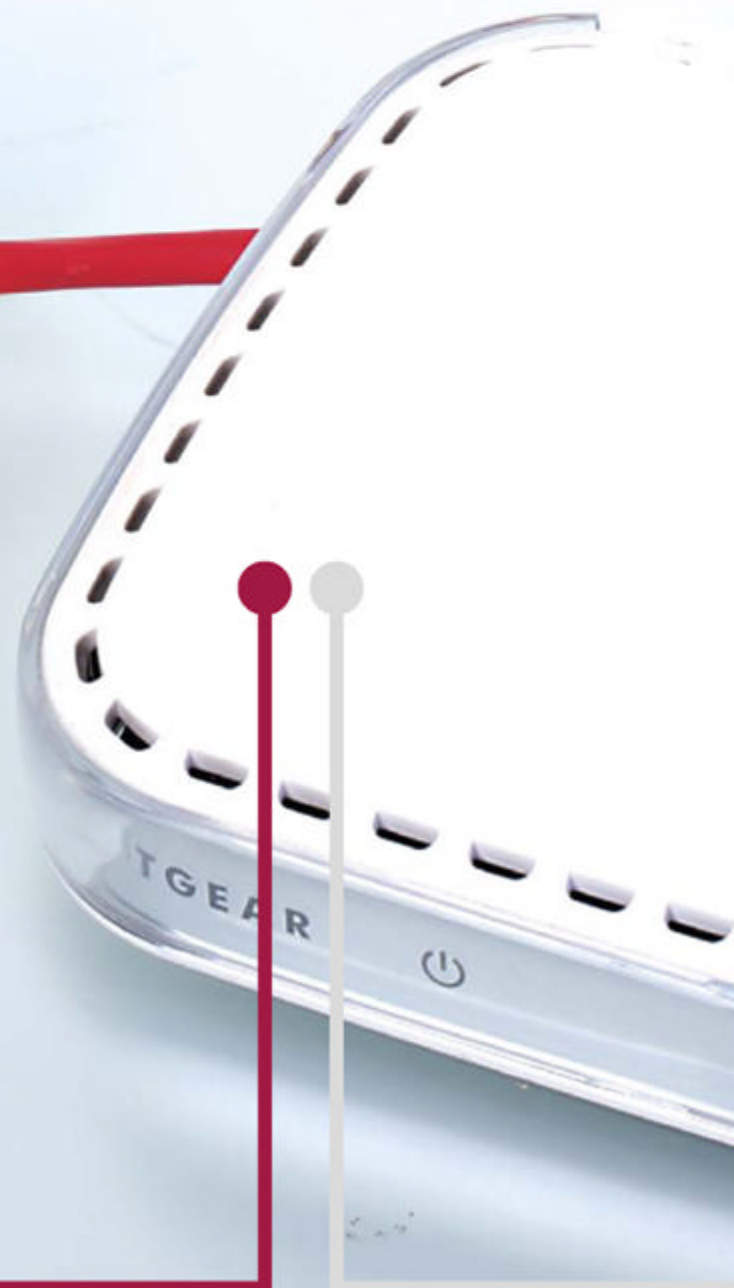


Browse privately with Onion Pi

Turn your Raspberry Pi into a highly secure and very portable router to keep your system safe and your browsing anonymous, wherever you are



“We install Tor and do some basic setup tasks so that it routes traffic properly, and securely. This will keep you anonymous online”





It's pretty simple to turn your Raspberry Pi into a portable wireless router. What if it's not enough to know you can search the web, though? What if you want to be wholly secure as you do it? Then it's time to upgrade the router with Tor to protect your privacy on the internet.

This 'Onion Pi', as dubbed by Adafruit, combines Raspbian and Tor to create and secure a wireless access point using just a Raspberry Pi. This project is fairly straightforward: after setting up the wireless access point, we install Tor and do some basic setup tasks so that it routes traffic properly, and securely. This will keep you anonymous online – a handy feature in a time of privacy concerns all around the web.

When the Pi is not connected to the internet, it should still function as a wireless router, allowing at the very least a wireless LAN in your location.



THE PROJECT ESSENTIALS

A Raspberry Pi

Raspbian

[raspberrypi.org/
downloads](http://raspberrypi.org/downloads)

**Compatible Wi-Fi
adaptor**

[adafruit.com/
products/814](http://adafruit.com/products/814)

01 Install Raspbian

Raspbian is the Raspberry Pi distro we'll be using for the Onion Pi. Download the zip file, extract the image and then apply it to an SD card using:

```
$ dd bs=4M if=[version number]-wheezy-  
raspbian.img of=/dev/[SD card location]
```

You can also use NOOBS to install Raspbian if you wish.

02 Set up Raspbian

Go through the initial Raspbian setup and make sure to turn on the SSH server, and to disable autoboot to desktop – this is unnecessary and will only use extra power. You can also tell it to fill up the rest of the card if there's room for it.

“Onion Pi combines Raspbian and Tor to create and secure a wireless access point”



03 Pi IP

We'll be accessing your Raspberry Pi via SSH to set it up. To do this we need to know its IP address – you can find it by typing `ifconfig` into the command line. Make a note of it and turn off your Pi.

04 SSH connection

Plug your USB wireless adapter into the Pi and then turn it back on. On another computer connected to the same network, open a terminal or type into the command line:

```
$ ssh [user]@[IP address]
```

Then enter the password for your Raspbian if it asks for it.

05 Install DHCP

To make life easier for any system connecting to the Pi access point, we need to install a DHCP server to it. We do this with:

```
$ sudo apt-get install hostapd isc-dhcp-server
```

DHCP will automatically assign IP addresses to network-attached devices, meaning you won't need static IPs.

06 Set up DHCP

Now we need to configure the DHCP server. Edit the configuration file with:

```
$ sudo nano /etc/dhcp/dhcpd.conf
```

And start by putting a `#` in front of the two option domain-name entries, then remove the `#` in front of authoritative, seven lines down.

“DHCP will automatically assign IP addresses to network-attached devices, meaning you won't need static IPs”

07 Server address

At the end of the configuration file, add the following:

```
subnet 192.168.42.0 netmask 255.255.255.0 {
range 192.168.42.10 192.168.42.50;
option broadcast-address 192.168.42.255;
option routers 192.168.42.1;
default-lease-time 600;
max-lease-time 7200;
option domain-name "local";
option domain-name-servers 8.8.8.8, 8.8.4.4;
}
```

Now save and exit.

08 DHCP server

Edit the server configuration files so that it's set to work in conjunction with the wireless adaptor:

```
$ sudo nano /etc/default/isc-dhcp-server
```

Scroll to INTERFACES and change it to:

```
INTERFACES="wlan0"
```

Below Adafruit sells an all-in-one Onion Pi pack (including a Ras Pi) for \$80: adafruit.com/products/1410

09 Incoming Wi-Fi

We now need to set up the Wi-Fi adaptor to be both static and accept incoming signals. First:

```
$ sudo nano /etc/network/interfaces
```

Put a # in front of iface wlan0 and following lines with wpa roam, iface default and any other affecting wlan0.



10 Static IP

Now give the wireless interface a static IP – after the line `allow-hotplug wlan0`, enter the following:

```
iface wlan0 inet static
    address 192.168.42.1
    netmask 255.255.255.0
```

Save and exit, and then set `wlan0`'s address with:

```
$ sudo ifconfig wlan0 192.168.42.1
```

11 WLAN creation

We need to create a new file that holds all the information for our wireless network. We are going to make it password protected so that only the people we want to can access it. To create the file, start with:

```
$ sudo nano /etc/hostapd/hostapd.conf
```

And then enter the text from the next step.

12 WLAN configuration

```
interface=wlan0
driver=rtl871xdrv
ssid=[access point name]
hw_mode=g
channel=1
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=[password]
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP
rsn_pairwise=CCMP
```

“We are going to make it password protected so that only the people we want to can access it”

13 Hostapd

After saving and exiting, we need to edit hostapd to point it to this new file. Open it with:

```
$ sudo nano /etc/default/hostapd
```

And then find the line `#DAEMON_CONF=""`. Remove the `#`, and change it to:

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

14 Network addressing

Setting up a NAT will allow multiple clients to connect.

To do this, run:

```
$ sudo nano /etc/sysctl.conf
```

And add to the bottom of the file:

```
net.ipv4.ip_forward=1
```

Save this, and then finish by running:

```
$ sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"
```

15 IP tables

Run the following three commands to make sure the internet connection is forwarded correctly:

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

```
sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

```
sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

16 Apply configuration

So that this still works after a reboot, type:

```
$ sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

Then add to the end of `/etc/network/interfaces`:

```
up iptables-restore < /etc/iptables.ipv4.nat
```

“Setting up a NAT (network address translator) will allow multiple clients to connect”

17 Wi-Fi final

Finally, set it up as a daemon so it runs at boot with the following commands:

```
sudo service hostapd start
sudo service isc-dhcp-server start
sudo update-rc.d hostapd enable
sudo update-rc.d isc-dhcp-server enable
```

And the wireless access point part will be finished.

18 Install Tor

After a reboot, we now need to install Tor. Do this simply with:

```
$ sudo apt-get install tor
```

Once it's installed, you will need to edit the Tor config file with:

```
$ sudo nano /etc/tor/torrc
```

Follow the next step to add all the necessary information to it.

Below You can download Tor to your other computers too

19 Tor configure

Put this below the FAQ comment:

```
Log notice file /var/log/tor/
notices.log
```

```
VirtualAddrNetwork
```

```
10.192.0.0/10
```

```
AutomapHostsSuffixes .onion,.
exit
```

```
AutomapHostsOnResolve 1
```

```
TransPort 9040
```

```
TransListenAddress
```

```
192.168.42.1
```

```
DNSPort 53
```

```
DNSListenAddress 192.168.42.1
```



20 Table flush

We now need to flush the current IP tables so that we can get the routing to go through Tor. First of all, do:

```
$ sudo iptables -F
```

```
$ sudo iptables -t nat -F
```

If you want to keep SSH open to connect remotely, you'll need to make an exception for that with:

```
$ sudo iptables -t nat -A PREROUTING -i wlan0  
-p tcp --dport 22 -j REDIRECT --to-ports 22
```

21 Reroute

Route all DNS traffic first, using:

```
$ sudo iptables -t nat -A PREROUTING -i wlan0  
-p udp --dport 53 -j REDIRECT --to-ports 53
```

And then route any TCP traffic with:

```
$ sudo iptables -t nat -A PREROUTING -i wlan0  
-p tcp --syn -j REDIRECT --to-ports 9040
```

22 Check and save

You can check the table setup with:

```
$ sudo iptables -t nat -L
```

If you're happy with it, save it to the NAT file like before with:

```
$ sudo sh -c "iptables-save > /etc/iptables.  
ipv4.nat"
```

“If you want to keep SSH open to connect remotely, you'll need to make an exception for that”



23 Logging

We should create a log file in case you need to debug later. To do this, use these three commands:

```
$ sudo touch /var/log/tor/notices.log
$ sudo chown debian-tor /var/log/tor/notices.log
$ sudo chmod 644 /var/log/tor/notices.log
```

You can also check it with:

```
$ ls -l /var/log/tor
```

24 Secure the router

Finally, we can activate the Tor service so that we can start using the access point securely with:

```
$ sudo service tor start
```

You can check this if you wish with:

```
$ sudo service tor status
```

To make it turn on at boot, you simple add it to rc.d with:

```
$ sudo update-rc.d tor enable
```

“We can activate the Tor service so that we can start using the access point securely with `sudo service tor start`”



A detailed LEGO Technic model of the droid R2-D2. The model is primarily white with blue and silver accents. It features a dome-shaped head with a large blue sensor, a smaller blue sensor, and a green light. The body is rectangular with various panels, vents, and mechanical details. The legs are thick and white, with blue joints. The model is standing on a wooden surface.





So what inspired you to build Pi2D2?

Growing up, I've always had a fascination for, and also a kind of a fear of, animatronics – you know, like the old Chuck E. Cheese-style robots and things. I was working with some people here in the barrio – we were restoring some old animatronics from a restaurant from the Eighties – and that's when the Raspberry Pi came out. So my first thought was that I can just use this to control the animatronics, but then I also had other ideas. I thought I'd really like to make an R2D2 that was controlled with the Raspberry Pi. I saw that someone else online had already beaten me to it – they'd made a voice-activated R2D2 using the Raspberry Pi, and I thought that was really cool – but then I thought 'What would I like to do with mine?'

I thought it would be kind of cool to be able to have direct control of him through a keyboard or a remote control, and then my wife had this old... like a Pico pocket projector that she bought off eBay that she wanted to use for a classroom, but she couldn't really use it. And I thought I could do something with this, so I had the idea of just having a remote control R2D2 that could move around and would have a little projector that could project images against the wall, and also use him as a soundboard, where I could put as many sound files as I wanted onto it and have him go around the office here at Telltale Games and harass people.

And then I saw another tutorial online, which was about a guy that made a project called Cambot, where he used a webcam mounted on an RC remote control car that was powered by the Raspberry Pi. So I was able to take what he did and apply it to my own project. So, you know, a lot of these elements already existed, and it was a matter of putting it together and making it work.



Andrew Langley

is a designer and programmer for Telltale Games. He was the lead designer on episode four of The Walking Dead and programmed Sam & Max: The Devil's Playhouse



What can Pi2D2 do, then? What features have you given him?

So it's got a little USB stick that you can put in and load as many WAV files, MP3 files, even movie files, as you want. Once that's put in, and you turn on the R2D2, he will join whatever Wi-Fi network you tell him to join. So then there's a webpage; if you go to the IP address of the R2D2, you're presented with a webpage, and it shows you a webcam and it shows you all the buttons for all the sounds that you put on your USB drive. You can hit keyboard shortcuts to play any sounds you want, and you can turn the projector on and off at will. You can play any video you want to from the webpage.

And that's basically what it does. Working at a videogame company, I used the traditional keyboard controls for a moving character – so it's WASD to move him around, and Space, and I can rotate his dome and look around and have him go up to people and talk to them, and kinda freak people out a bit at work.

Have you thought about making a smartphone or tablet interface for him?

Yeah I toyed with that – at the time when I was developing it, I didn't really find a good library that did touch controls that well, but I think things have come out since then that do provide that, so that's definitely something [to consider]. I have a lot of upgrades that I want to do to it and that's one. I also want to... it's kind of tricky to have him join whatever Wi-Fi network that you're on, so I think I'm going to make it so that instead, he's his own little Wi-Fi hotspot, so any computer can just log into him and then be presented with the webpage to control him. So that's definitely the thing I want to do next.

If you like

You can check out the original hacked R2D2, with facial recognition that inspired Andrew over at:

bit.ly/1kmrrRL

Further reading

For the full list of hardware components used to make Pi2D2, see Andrew's blog:

langleycreations.com/blog/?p=17



So what's powering it all?

It's using a LiPo battery – actually two LiPo batteries – and one of them controls the Raspberry Pi, which is running the Raspbian Linux image, and then the other battery controls the DC motors for rotating the dome and the legs. So when you switch on the Raspberry Pi (and it usually takes about twenty seconds to boot up), it will join your Wi-Fi network automatically. Currently what I have him doing is, once he's logged onto the network, he pings a PHP script on my server which provides you with his IP address, so then you can go to your browser, enter that IP address and you're presented with the webpage and webcam and the buttons, and you can control him however you want.

Below As well as Princess Leia's plea for help, Pi2 can play custom sounds and *Mario* game footage



You seem to have some storage compartments built in as well?

Yeah the original toy comes with two little storage compartments on the side. One has a little cupholder that extends, and you can put in a glass of beer or something, and the other one was left empty. It was important to me for whatever reason to keep the cupholder intact, which I was able to do. And then the other side, when you open it, that's where the charging ports are for the projector and the battery for the motors, and that's also where you insert your USB drive that has all the sounds and videos and that. I have it where if you name the files in a certain way – like 1.wav, 2.wav – those keys on your keyboard will also play it, if you want to have super quick control.

How much time do you think it took?

It was a pretty long project. I didn't work on it full time, obviously, but I probably worked on it over a period of six months, and most of the time was writing the software. A lot of the software was written in Python – like the controls for the webcam, the soundboard and everything – so most of the time was getting the software running and getting the kinks worked out. Like where if it loses a Wi-Fi connection it tries to rejoin and things like that. So, yeah, I definitely want to revisit it, and obviously the second time round you can do it a lot better than you did the first, so I'd like to go back.

Maybe you could introduce C3PO for the next one...

You know, here at Telltale we're working on a *Borderlands* game and I thought it'd be really fun to make a *Borderlands* robot that would go around the office and harass people. So, you know... you never know.

“You know, here at Telltale we're working on a *Borderlands* game and I thought it'd be really fun to make a *Borderlands* robot that would go around the office and harass people”



Make a complete media centre

Turn your Raspberry Pi into the perfect media centre with some simple software and tweaks





Your Raspberry Pi is uniquely suited to being a home theatre PC as it's a great low-power device that you can happily have running alongside your main computer, acting as a media server for both your desktop and your mobile devices through remote apps, not to mention those of any friends who come visiting.

There are a few ways to use the Raspberry Pi as an HTPC, with some of the best being XBMC-powered distros. OpenELEC is a great, lightweight way to use the Pi for that sole purpose, but we're going to cover the more customisable Raspbmc, based on Debian.



THE PROJECT ESSENTIALS

Internet connection

External hard drive

VESA mount

HDMI cable

01 Install Raspbmc

Raspbmc has a downloadable installer that always gets the latest image for your Pi. Create a new directory and then download it with:

```
$ wget http://svn.stmlabs.com/svn/raspbmc/  
release/installers/python/install.py
```

Make it executable with:

```
$ chmod +x install.py
```

And then run it with:

```
$ sudo python install.py
```

Follow the on-screen instructions and it will then install onto your Pi.

02 Setup

By installing Raspbmc this way, the distro will then download the latest image and updates to run on your Pi on first boot. This will take a while, and involve a couple of restarts, but will only happen the first time. Select your language in XBMC and you're done.

“OpenELEC is a great, lightweight way to use the Pi as a HTPC, but we're going to cover the more customisable Raspbmc”



03 Network sharing

You can add file sources to XBMC by first of all going to Video, then Add Source. Go to Browse and you can add local or networked files. For network shares, choose either UPnP or SMB shares to see what's visible on the network, or 'Add network location' if you know the direct path. This works the same for the Music and Pictures areas too.



Above XBMC (Xbox Media Centre) was recently renamed Kodi to move the project on from its console-linked past

04 Scrape your media

After selecting a source, you can choose what service to scrape information from. This gives you more info on the files you're watching, such as episode summaries and proper names. Make sure your films have the year appended to them, and that your TV shows do too.

05 Get some apps

On any content tab, go to Add-Ons. It will display the apps you already have, which is none by default. By selecting Get More, the full list of available apps will be shown and you can install them with one button click from there.

06 Web remote

In Settings, go to Services and then Webserver. Enable it and you will now be able to control your Raspberry Pi using a web browser. To access the web browser

“For network shares, choose either UPnP or SMB shares to see what's visible on the network, or 'Add network location' if you know the path”

remote, enter the IP address – found in System Info under the home screen's Settings tab – into your browser, followed by ':80'.

07 Android Remote

There's an official Android Remote app on the Google Play Store that can control XBMC once the web server is enabled. Download it to your phone and launch it. Go to the Settings, add new host, and enter the IP address and port (80 by default) like in the previous step.



Above You can control your start-up, standby and power-down routines in the Settings menu

08 Using CEC

If your TV supports it, CEC can be used over HDMI to control your XBMC-powered Pi. CEC should be enabled by default, which means you can control your entire setup with only one remote control. Refer to the XBMC wiki to see if your TV supports CEC, and how you can take advantage of it: bit.ly/18kdVHf.

09 Traditional remotes

XBMC supports a large number of universal remotes that include, or support, USB IR receivers. One of the best devices for this is the FLIRC, a fully programmable USB IR receiver that allows you to use any remote with the Pi.



Build a VoIP server

Use your Raspberry Pi as a voice over IP server,
perfect for phone calls in the home or a small
business





Especially if you're travelling abroad and looking to avoid some nasty roaming charges, voice over IP is a great way to make a phone call over the Internet. RasPBX is a project that brings Asterisk – industry-standard voice over IP software – to the Raspberry Pi. There are VoIP clients for many platforms, including hardware that converts VoIP to a conventional telephone line. Each client gets a phone number and can call other clients. You can also have conference calls – perfect for meetings over the phone. After this tutorial, you could also share the VoIP server over the Internet, allowing remote clients to connect.



THE PROJECT ESSENTIALS

A router or switch

Smartphones,
computers, or VoIP
phones

Powered USB hub

Latest RasPBX image

01 Initial setup

RasPBX is good to go as soon as the image is flashed to your SD card: there is a web interface for configuration, and SSH is included for remote login. As this will be a server, we'll log in with the user 'root' and password 'raspberry' and change the IP address to a static one so we always know where it is on the network.

```
root@raspbx:~# ip addr show dev eth0 | grep
inet
inet 172.17.173.94/24 brd 172.17.173.255 scope
global eth0
inet6 fe80::ba27:ebff:fef3:9016/64 scope link
root@raspbx:~# ip route | grep default
default via 172.17.173.1 dev eth0
root@raspbx:~# cat /etc/resolv.conf
nameserver 127.0.0.1
nameserver 8.8.8.8
nameserver 8.8.8.4
```

“There are VoIP clients for many platforms, including hardware that converts VoIP to a conventional telephone line”



02 Assign a static IP address

```
iface eth0 inet static
    address 172.17.173.94
    netmask 255.255.255.0
    network 172.17.173.0
    broadcast 172.17.173.255
    gateway 172.17.173.1
```

Now that we have the network configuration, we can assign a static IP address. Open `/etc/network/interfaces` in an editor such as nano, and change the line:

```
iface eth0 inet dhcp
```

...to a configuration similar to our expert's. We don't have to worry about DNS as Google's nameservers are being used. You can load the new network config using

```
sudo /etc/init.d/networking restart
```

03 The web Interface

Type the static IP address of your Raspberry Pi into a web browser. You'll be taken to the FreePBX web interface for configuring Asterisk. Click the FreePBX Administration button and use the username 'admin', and password 'admin'. The User Control Panel allows users to listen to their voicemails.

Below FreePBX is a free GUI tool for working with Asterisk



FreePBX
Administration



User Control Panel



Get Support

04 Add extensions

Each device that will be connected needs its own extension. To add an extension, hover over the Applications tab, then select Extensions. Select Generic SIP Device as the device type and click Submit. The User Extension is the number to call to get to that device. The Display Name can either be the name of a person or just the same as the User Extension. The only other thing that you need to fill in is the secret, which is a passphrase that allows the device to connect. One of these will have been generated, but you can change it if you like. Scroll to the bottom and click Submit. Add as many extensions as you like.

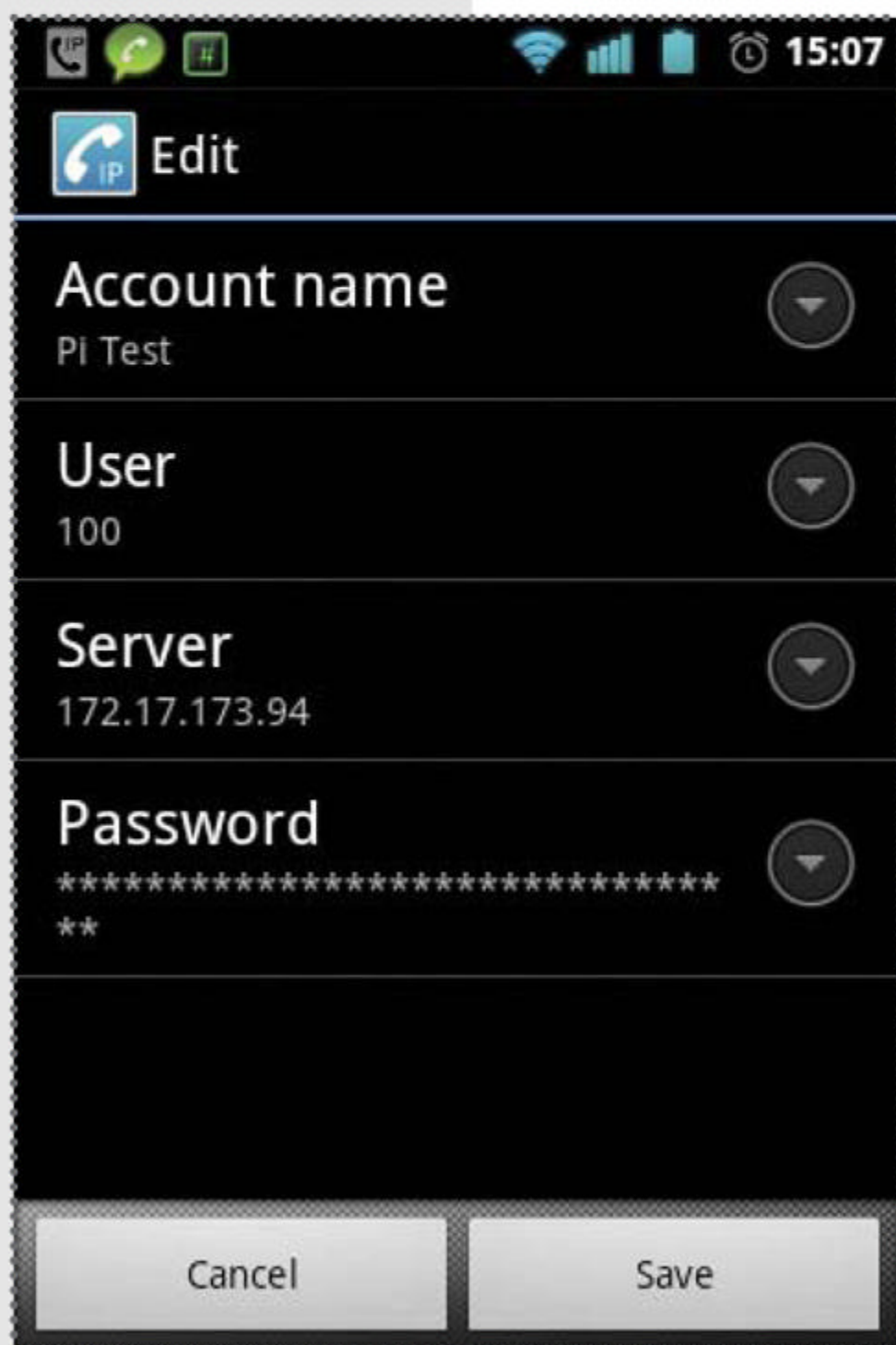
Below On Android, you can use your VoIP server to make calls via CSipSimple

05 Add a conference extension

Go to the Applications section and select Conferences. Choose a number and name for the conference, and a PIN number if you'd like users to require a PIN to join. Click Submit Changes once you're done.

06 Apply config

Changes such as adding extensions and conferences are not made until the Apply Config button is pressed, so be sure to do that before closing the web interface.



The screenshot shows an Android application interface with a dark theme. At the top, there's a status bar with icons for IP, a green speech bubble, a green square with a white 'H', Wi-Fi, cellular signal, battery, and a clock showing 15:07. Below the status bar is a header bar with a blue circular icon containing a white 'IP' and the word 'Edit' in white. The main content area consists of four rows, each with a label, a value, and a circular dropdown arrow icon on the right. The rows are: 'Account name' with the value 'Pi Test'; 'User' with the value '100'; 'Server' with the value '172.17.173.94'; and 'Password' with the value '*****' and two asterisks below it. At the bottom of the screen are two large, light gray buttons labeled 'Cancel' and 'Save'.

07 Trying it out

If you're using an Android phone, we recommend CSipSimple, which can be found on Google Play. We recommend Linphone for Linux clients. We'll use two Android phones as an example. When adding an account in CSipSimple, scroll down to the Generic wizards section and select Basic. Then fill in the information as shown in the image (previous page). Once you have two devices set up, try calling between the two.

08 Trying a conference call

Call the conference number that you set up from one of the devices. You'll need to enter the PIN code you set followed by the # key. You'll be told that you are the only person in this conference, and will hear a notification whenever anyone else joins and leaves the conference.

“If you're using an Android phone, we recommend CSipSimple. We recommend Linphone for Linux clients”



“You’ll no longer have to struggle to share files between computers using removable media or Dropbox”





Having a central file server in your house will make life much easier. You'll no longer have to struggle to share files between computers using removable media, or rely on off-site solutions such as a Google Drive or Dropbox account, because you can instead just save everything in your own mini-cloud on your home network. It's an inherently more secure option than third party clouds too, since there's no middleman, and the Raspberry Pi is perfect for this!



THE PROJECT ESSENTIALS

A router or switch

External storage
(optional)

Powered USB hub

Latest Raspbian image

01 Installing the required software

Log into the Raspbian system with the username Pi and the password 'raspberry'. Get the latest package lists using the command `sudo apt-get update`. Then install the required packages with the command `sudo apt-get install samba`. Samba will be used to serve files and also comes with software to advertise file shares on the network.

02 Connect external storage

If you have any external storage, you'll need to prepare it for use with Linux. Note that you will only be able to read the storage device on other Linux computers unless you reformat it. If you connect the device, you'll be able to use the `dmesg` command to find out what the device is called. (Piping that into `tail -n 3` prints the last three lines.) It's probably `/dev/sda`.

```
pi@raspberrypi ~ $ dmesg | tail -n 3
[ 1707.371370] sd 1:0:0:0: [sda] No Caching
mode page present
[ 1707.371403] sd 1:0:0:0: [sda] Assuming drive
cache: write through
[ 1707.371422] sd 1:0:0:0: [sda]
```



Left People on your home network will be able to sign in to access the file server

03 Format external storage

Use Parted to create a new partition table on your external storage device and make a single partition the full size of the device. Then run partprobe to reload the partition table, and make an ext4 file system on the new partition. This will destroy all existing data on the disk.

```
pi@raspberrypi ~ $ sudo parted /dev/sda
(parted) print
Disk /dev/sda: 16.1GB
(parted) mktable msdos
Warning: The existing disk label on /dev/sda
will be destroyed and all data on this disk
will be lost. Do you want to continue?
Yes/No? Yes
(parted) mkpart
Partition type?  primary/extended? primary
File system type?  [ext2]? ext2
Start? 0GB
End? 16GB
```

“Use Parted to create a new partition table on your external storage device and make a single partition the full size of the device”

```
(parted) quit
pi@raspberrypi ~ $ sudo partprobe
pi@raspberrypi ~ $ sudo mkfs.ext4 /dev/sda1
```

04 Mount external storage

We're going to move /home onto our external storage. We'll do that by mounting it on /mnt/storage, moving the files from /home to /mnt/storage, and then mounting the partition on the now empty /home.

```
pi@raspberrypi ~ $ sudo su
root@raspberrypi:/home/pi# cd /
root@raspberrypi:/# mkdir /mnt/home
root@raspberrypi:/# mount /dev/sda1 /mnt/home/
root@raspberrypi:/# mv /home/* /mnt/home/
root@raspberrypi:/# umount /mnt/home/
root@raspberrypi:/# rmdir /mnt/home/
root@raspberrypi:/# echo '/dev/sda1 /home ext4
defaults 0 1' >> /etc/fstab
root@raspberrypi:/# mount -a
root@raspberrypi:/# mount | grep sda1
/dev/sda1 on /home type ext4
(rw,relatime,data=ordered)
root@raspberrypi:/# ls /home
lost+found pi
root@raspberrypi:/# exit
```

05 Create users for Samba

We now need to create some users to log into Samba shares with. Do the following steps for as many users as you'd like to have on the server. Only do the usermod step if you want the user to be able to use a shared directory for all users.

“We'll move /home onto our external storage by mounting it on /mnt/storage, moving the files from /home to /mnt/storage, then mounting the partition on /home”


```
pi@raspberrypi ~ $ sudo adduser liam
Adding user 'liam' ...
Adding new group 'liam' (1002) ...
Adding new user 'liam' (1001) with group
'liam' ...
Creating home directory '/home/liam' ...
Copying files from '/etc/skel' ...
sh: 0: getcwd() failed: No such file or
directory
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for liam
Enter the new value, or press ENTER for the
default
    Full Name []: Liam Fraser
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
pi@raspberrypi ~ $ sudo usermod -a -G
sambashare liam
pi@raspberrypi ~ $ sudo pdbedit -a -u liam
new password:
retype new password:
Unix username:      liam
Full Name:          Liam Fraser
Home Directory:     \\raspberrypi\\liam
```

“Only do the usermod step if you want the user to be able to use a shared directory for all users”

06 Create a shared directory

We're going to create a directory that any users in the sambashare group can access and change. Setting



the setgid permission on a directory (chmod g+s) causes new files and subdirectories created within it to inherit its group ID. This means that each user in the sambashare group can access and change the files.

```
pi@raspberrypi /home $ sudo mkdir /home/  
allusers
```

```
pi@raspberrypi /home $ sudo chown  
root:sambashare /home/allusers/
```

```
pi@raspberrypi /home $ sudo chmod 770 /home/  
allusers/
```

```
pi@raspberrypi /home $ sudo chmod g+s /home/  
allusers/
```

07 Configuring Samba

Open /etc/samba/smb.conf in an editor using sudo. Scroll down to the Authentication section and uncomment the security = user line. Then scroll down to the Share Definitions section and change read only to no in the [homes] section. Finally, scroll down to the bottom of the file and then add in a section for the shared folder.

```
[allusers]
```

```
comment = Shared
```

```
Folder
```

```
path = /home/allusers
```

```
read only = no
```

```
guest ok = no
```

```
browseable = yes
```

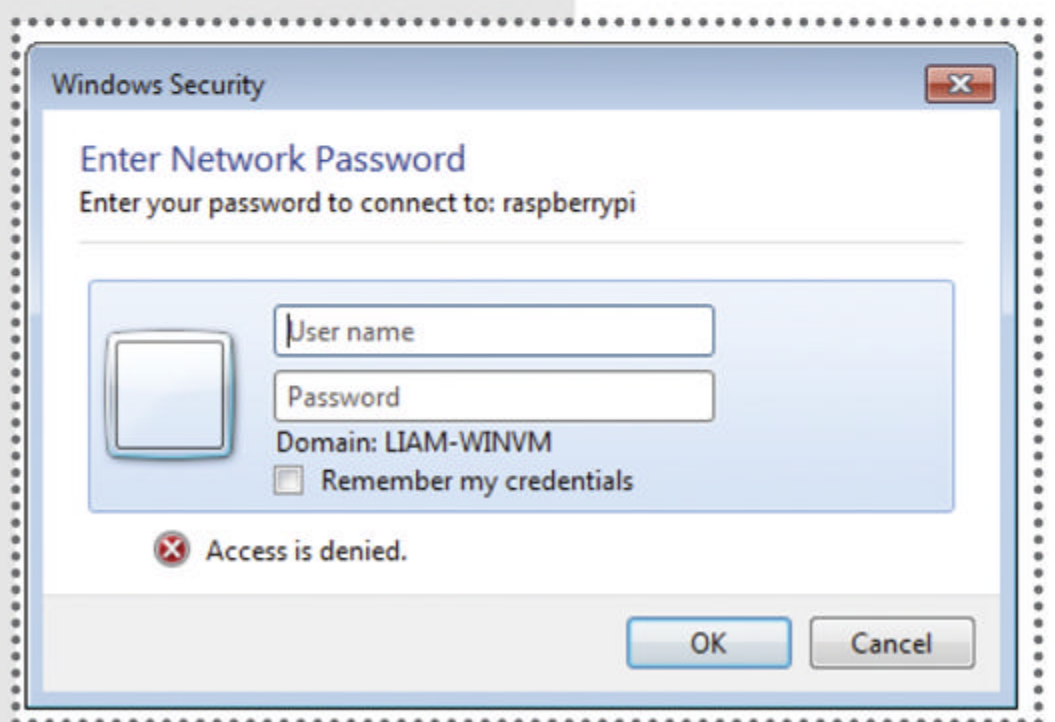
```
create mask = 0770
```

```
directory mask = 0770
```

Restart Samba with

```
sudo /etc/init.d/samba  
restart
```

Below Consider setting up a guest account for unexpected users



08 Connecting to the shares

The Raspberry Pi will advertise its shares automatically. Each user will get access to their home folder, as well as the allusers directory. If the user isn't in the sambashares group, they won't be able to get it. The syntax for accessing Samba shares is as follows:

Windows: `\\raspberrypi\share_name`

Mac: `smb://raspberrypi/share_name`

Linux: `smb://raspberrypi/share_name`

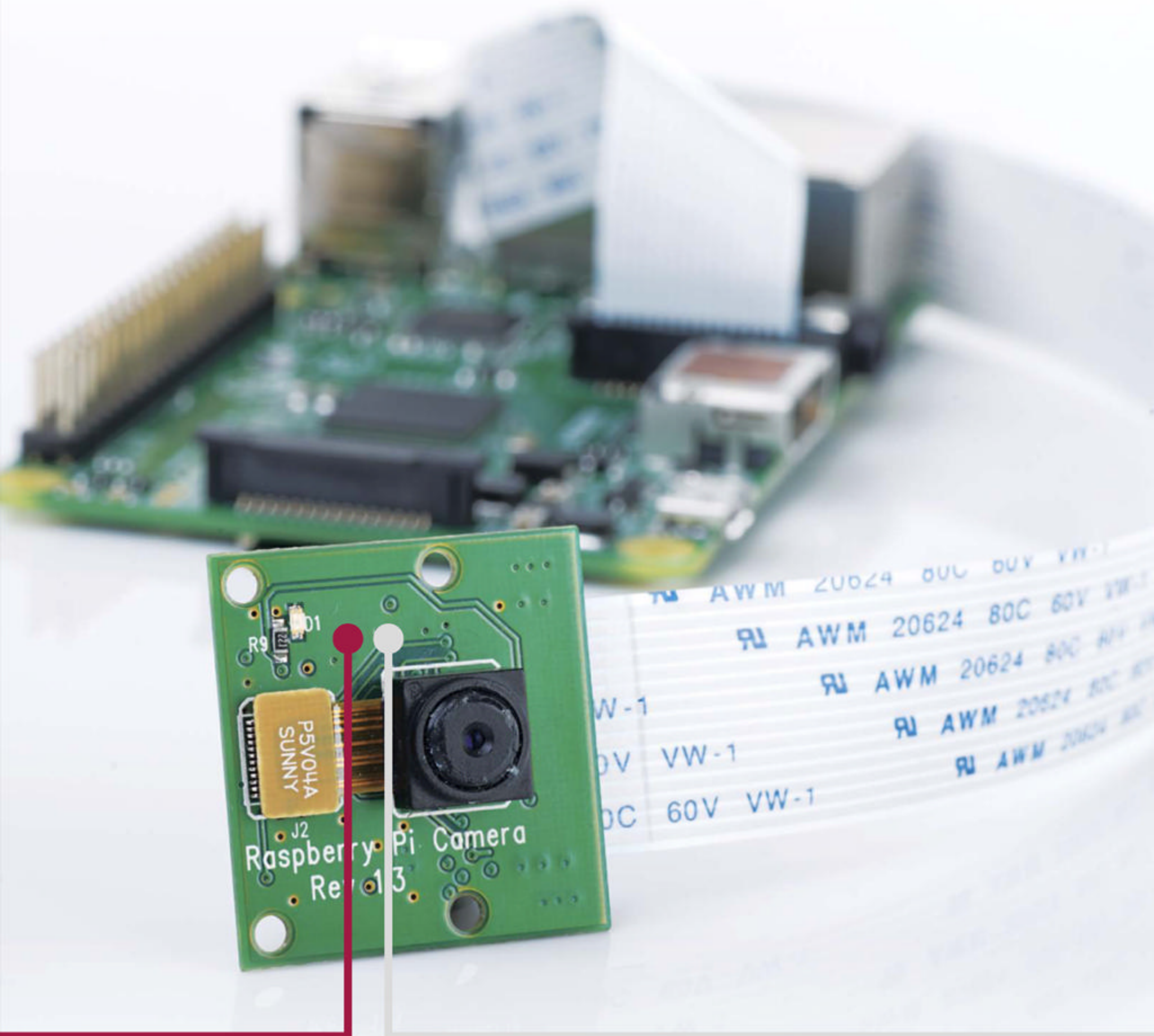
The share name is optional – you should be able to browse for them. In fact, you shouldn't even need to use these; you just need to go to the Network section of your file manager and select 'raspberrypi'.

“Each user will get access to their home folder, as well as the allusers directory. If the user isn't in the sambashares group, they won't be able to get it”



Slow-motion video

Use the latest Raspberry Pi firmware to record slow-motion videos with the camera module





The Raspberry Pi camera has been an interesting little add-on for the Raspberry Pi, allowing people to have slightly more control when it comes to projects that require pictures and video. The functionality of the camera is always being slightly enhanced and a relatively new function has made its way to the camera: slow-motion video.

With this feature you can create short clips as slow as 33 per cent normal speed, with all the same control as you have with the normal camera functions, and it's a great way to reinvigorate your existing camera module projects. At the time of writing, our favourite PiCamera Python module doesn't include the ability to access the slow-motion functions.



**THE PROJECT
ESSENTIALS**

Latset Raspbian
[raspberrypi.org/
downloads](https://raspberrypi.org/downloads)

Camera board
Internet connection

01 The firmware updater

The first thing you need to do is make sure you update the firmware on your Raspberry Pi. Boot into Raspbian and open the LXTerminal. To install the firmware updater, simply type in the following command. For newer versions of Raspbian, it will likely have already been installed.

```
$ sudo apt-get install rpi-update
```

02 Update the firmware

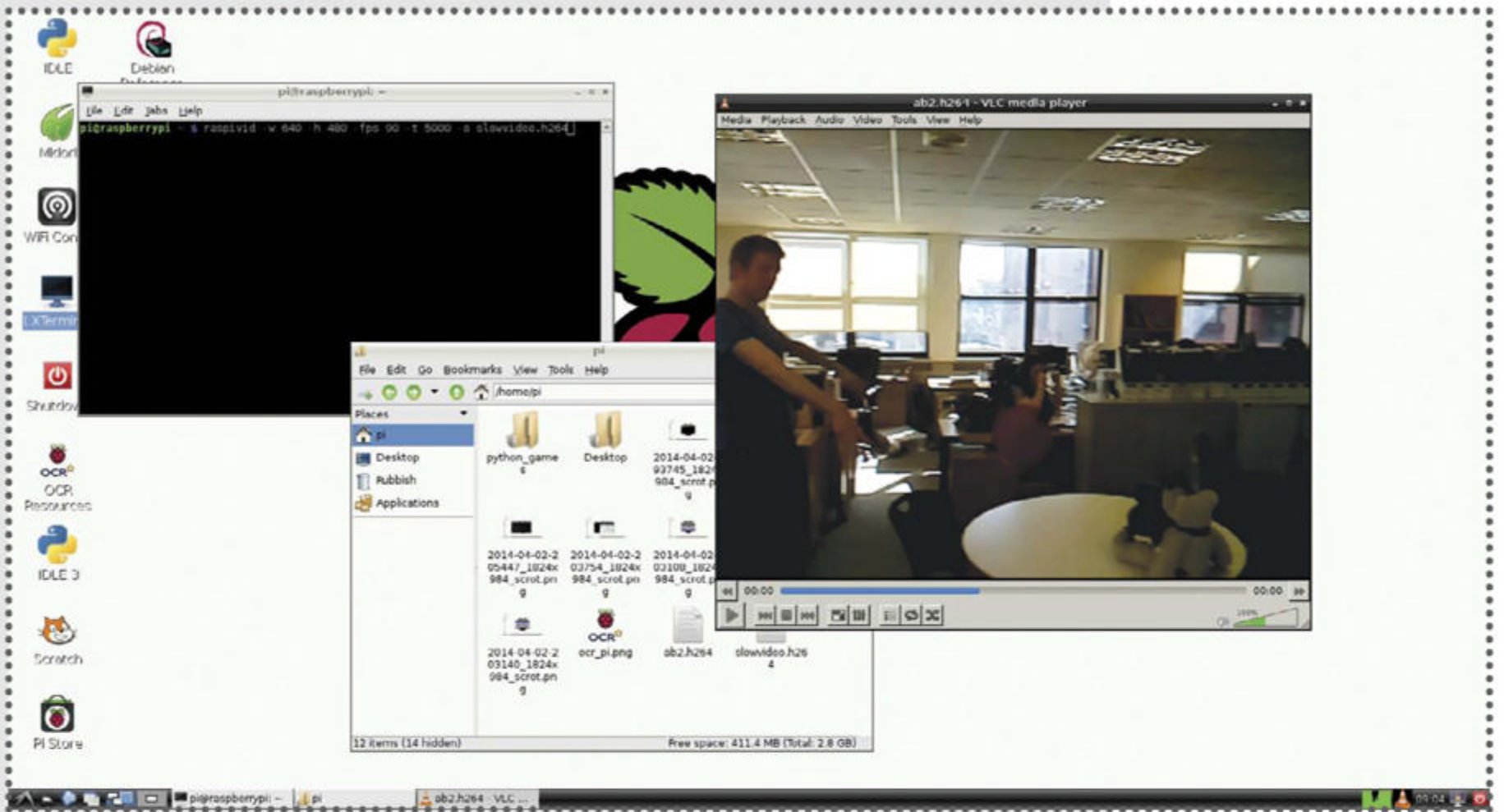
Updating the firmware is very straightforward; back in the terminal just type in the following:

```
$ sudo rpi-update
```

...and it should automatically download and install all the necessary files. Once this is complete, reboot your Raspberry Pi to activate the new firmware

“You can create short clips as slow as 33 per cent normal speed, with all the same control as you have with the normal camera functions”





Above A number of new video modes with higher resolutions were added last year

03 Alternate updating

Problems have been known to occur with the firmware update and there are two main ways you can try and fix them. First of all, try updating Raspbian with apt-get update followed by apt-get upgrade. If that method doesn't work, try using the firmware updater like so:

```
$ sudo UPDATE_SELF=0 rpi-update
```

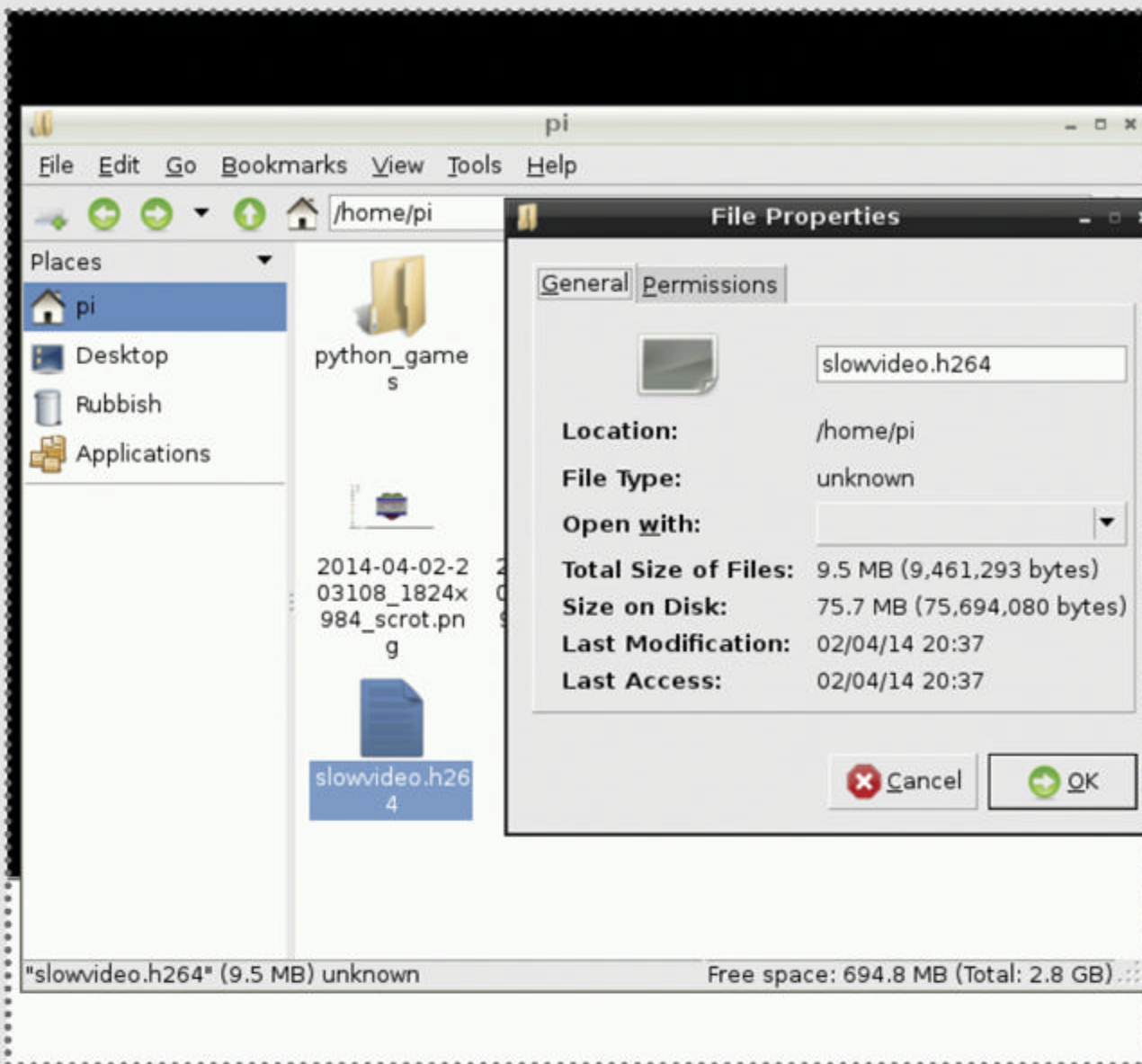
04 Camera setup

Just in case you're using a new install of Raspbian, you'll need to make sure that the camera module is enabled. Make sure that you've plugged in the camera while it's off, and then in the LXTerminal use:

```
$ sudo raspi-config
```

Go to the Enable Camera option and enable it. You may need to restart your Pi in order for it to actually take effect.

Left Raspistill creates a H264 file, which can be opened using the VLC media player



05 First tests

We can now do a quick test of the slow-motion capabilities by entering the following:

```
$ raspivid -w 640 -h 480 -fps 90 -t 5000 -o slowvideo.h264
```

We've told it to make a video at 640 x 480 resolution, to film it at 90 frames per second and to do so for five seconds. This is the maximum resolution for this speed of video.

06 Other recording modes

As well as being able to record video at 90 frames per second, you can also go down to 60. You cannot increase the resolution though, so it's stuck to 640 x 480 for both 90 fps and 60 fps video for the time being.



07 Playing it back

Although your Pi has the means to record the video, it can't play it back. You can either grab the video from the SD card, or install VLC using the following command, which should have no trouble viewing the video file.

```
$ sudo apt-get install vlc
```

08 Other uses

If you do plan to use the slow-motion function, obviously you can use it as intended. However, you can also use it to create high frame rate videos. Video editors such as Kdenlive will let you increase the playback speed, and you can also view them in VLC at normal speed.

09 Final warnings

Doing these slow-motion videos – even at the low resolution we're using – is taxing on the processor. Try to only capture short clips while using it and definitely not a lot of them in rapid succession.

“Video editors such as Kdenlive will let you increase the playback speed, and you can also view them in VLC at normal speed”



Talking Pi

Join the conversation at...



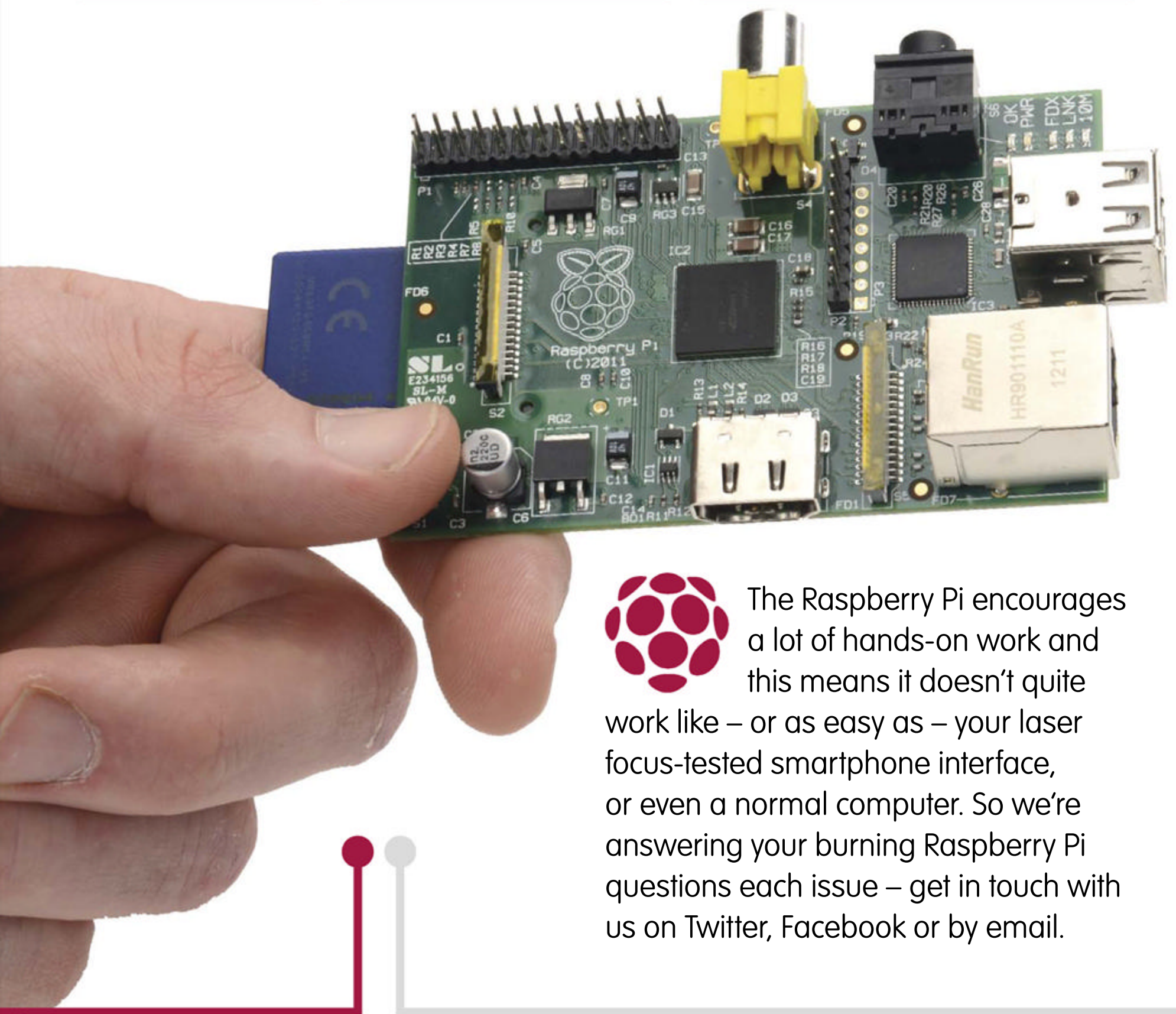
@linuxusermag



Linux User & Developer



RasPi@imagine-publishing.co.uk



The Raspberry Pi encourages a lot of hands-on work and this means it doesn't quite work like – or as easy as – your laser focus-tested smartphone interface, or even a normal computer. So we're answering your burning Raspberry Pi questions each issue – get in touch with us on Twitter, Facebook or by email.

Is it worth me getting a second Raspberry Pi?
Richard via Facebook

This depends entirely on the context – why do you think you need a second one? There are certainly applications for your Raspberry Pi where you'd want to basically just leave it alone, like a file server or private wireless router. A second one could help you just play around with the Pi without interrupting its service.

There are also a few ways to use two Raspberry Pi's in conjunction with each other, such as a Raspberry Pi super computer or similar. They're pretty cheap anyway, so if you have a project in mind for an extra one, go ahead.



Follow @LinuxUserMag on Twitter. Search for the hashtag #RasPiMag

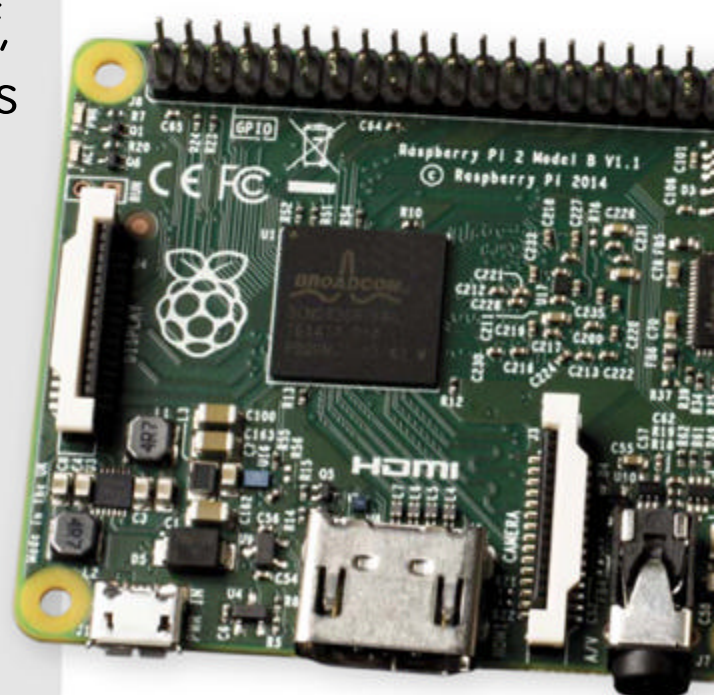
Win a new Raspberry Pi 2

The Raspberry Pi Foundation has just announced the brand new Raspberry Pi 2. Available to buy now and shipping with an updated version of Raspbian, the Pi 2 is the next evolution of the standard Model B.

The key difference is the brand new BCM2836 processor, which uses an ARMv7 core and is now six times more powerful than the old Pi. >>

Are there any switches for the Raspberry Pi to turn it on and off again?
Amy via Twitter

There are some homebrew methods that plug into the Raspberry Pi and allows for resets and switching on and off, but one of the simplest methods is a simple USB switch that can be bought off many online retailers, specialist or otherwise. It simply stops the power flow along the USB cable, allowing you to turn a Pi on and off again without needing to play about with the power lead or inside the command line. Check out adafruit.com/product/1620.



Is it fine to leave my Raspberry Pi without a case?

Ted via email

The Raspberry Pi is a sensitive piece of electronics. That being said, it's also quite hardy, so leaving it out every now and then is probably not going to break it unless you put it somewhere really unsuitable. It is preferable to keep it in a case though, if your Pi is for general use in various different projects, as it also stops you from putting pressure on different parts of the board as you plug and unplug different cables and devices.



Will SteamOS work on my Raspberry Pi?

Laura via Facebook

The short answer is: no. Valve's SteamOS, while definitely Linux-based, does not have a version available for the Raspberry Pi. There are a few hurdles in the way as well – firstly it's built for normal PCs, which have a different type of processor, and there's no real way to make it work on the Raspberry Pi's processor without any of the source files. Secondly, it also requires a lot more power than the Raspberry Pi can deliver, even if you're doing the streaming thing from a PC already running Steam. The Raspberry Pi was not really built for games unfortunately, so Steam OS is off the table.



Win a new Raspberry Pi 2

The Pi 2 will work with all your existing projects and is fully backwards compatible on the software side, so it's the perfect upgrade

We have five of these brilliant boards to give away, and for a chance to win one of them all you need to do is pay a visit to the webpage below and then answer one simple question.

Enter the Raspberry Pi 2 competition:
bit.ly/1CGPjUE





Next issue

🍷 Get inspired 🍷 Expert advice 🍷 Easy-to-follow guides



Build a

PI VIDEO PLAYER

Plus

Code your own
Twitter bot & more

Get this issue's source code at:
www.linuxuser.co.uk/raspicode